A user-centred design and evaluation of IR interfaces

S. M. Zabed Ahmed, Cliff McKnight and Charles Oppenheim
Journal of Librarianship and Information Science 2006 38: 157
DOI: 10.1177/0961000606063882

A user-centred design and evaluation of IR interfaces

S.M. ZABED AHMED, CLIFF McKNIGHT*
and CHARLES OPPENHEIM

S.M. Zabed Ahmed is an Associate Professor in the Department of Information Science and Library Management, University of Dhaka, Bangladesh. He specializes in usability studies with Web-based IR systems. He is particularly interested in exploring the factors affecting end-users' searching of IR systems and in adopting a user-centred approach in designing IR interfaces.

Cliff McKnight has a Chair in Information Studies in the Department of Information Science, Loughborough University. Before joining the Department in 1994, he was previously with the University's HUSAT Research Centre. His research interests revolve around human use of information resources. He is Editor in Chief of the Journal of Digital Information.

Charles Oppenheim has been Professor of Information Science at Loughborough University since 1998. Prior to that, he held a variety of posts in academia and the electronic publishing industry, working for International Thomson, Pergamon and Reuters at various times. He is also Project Director of LISU. Charles is an Honorary Fellow of the Chartered Institute of Library and Information Professionals. He is a member of the Legal Advisory Board of the European Commission. He was the Specialist Advisor to the House of Lords' Inquiry into the Information Superhighway. He is a regular contributor to conferences and to the professional and scholarly literature, and is on the editorial board of a number of professional and learned journals, and of the Annual Review of Information Science and Technology.

Address: Department of Information Science
Loughborough University
Loughborough
Leicestershire LE11 3TU
United Kingdom
Tel: +44 (0) 1509 223061
E-mail: c.mcknight@lboro.ac.uk

This paper presents a user-centred design and evaluation methodology for ensuring the usability of IR interfaces. The methodology is based on sequentially performing: a competitive analysis, user task analysis, heuristic evaluation, formative evaluation and a summative comparative evaluation. These techniques are described, and their application to iteratively design a prototype IR interface, which was then evaluated, is described. After each round of testing, the prototype was modified as needed. The user-centred methodology had a major impact in improving the interface. Results from the summative comparative evaluation suggest that users' performance improved significantly in our prototype interface compared with a similar competitive system. They were also more satisfied with the prototype design. This methodology provides a starting point for techniques that let IR researchers and practitioners design better IR interfaces that are both easy to learn to use and remember. The paper concludes with some principles of interface design for IR systems.

KEYWORDS: HCI; heuristic evaluation; interface design; task analysis

1. INTRODUCTION

There has been a general lack of attention given to IR interfaces by database vendors. Most IR systems in the 1970s and 1980s used structured command mode interfaces. The major disadvantage of such command systems is the fact that the users must be quite familiar with the command language of the system in order to use it effectively. Online searches at that time required skilled intermediaries because of the complexity of the user interfaces. In the 1980s, several IR systems introduced simplified versions of their user interfaces for direct end-user searching, using features such as simplified commands, menu-selection, form fill-in and direct manipulation interfaces. Many of these interfaces, however, had not been well accepted, mainly because the underlying systems were still difficult for end-users to use successfully in online searching. In the 1990s, IR system development was influenced by rapid technological advancements in both hardware and software. The development of Web-based access to IR systems in the mid-1990s has significantly changed end-users' access to such systems. These include a wider and cheaper access to a variety of IR systems and improved user interfaces and functions. Despite all these changes and improvements over the last three decades, studies have showed that IR systems are still difficult to learn, use and remember.

A number of attempts to investigate end-user IR searching have been reported in the literature. Even though the methodologies and approaches used in these studies differ considerably, the results consistently show the difficulties end-users have using online databases, the inefficiency of their

*To whom all correspondence should be addressed.

Vol 38 (3): 157–172
[DOI: 10.1177/0961000606063882]

searching techniques and the poor results they get (Fenichel, 1981; Penhale and Taylor, 1986; Sutcliffe at al., 2000; Ahmed et al., 2004). Even studies that reported a high-level of users' satisfaction (e.g. Sullivan et al., 1990) noted that end-users rely on overly simplistic searches, make frequent errors, and fail to obtain satisfactory results. It is agreed that this failure is caused not only by the lack of their search skills or training but also by the inadequacies of the search interfaces. Some studies demonstrated an awareness of the problems, but they merely suggest the need to improve the IR user interface designs and fail to provide specific recommendations (e.g. Hsieh-Yee, 1993; Barker, 1997; Zhang and Chignell, 2001).

The user interface of an IR system is an important feature that impacts on the users' performance and satisfaction with that system. The Web-based interfaces to IR systems must complement a variety of users' individual differences, their cognitive abilities and task requirements. Although designers follow style guides and de facto standards in designing IR interfaces, there is, however, no guarantee that an IR interface will attain high quality by these means only. IR interface designs need to be user-centred in order to support users' interactive information searching. IR researchers are becoming aware of traditional human-computer interaction (HCI) usability efforts and beginning to apply and expand upon those methods for designing IR interfaces. Prototyping, testing and iterative design are key activities in user-centred design. A few efforts have been reported to date (Mulhem and Nigay, 1996; Petrelli et al., 2004), however, user-centred design and evaluation in IR as a practice still lags far behind what is needed. There have been some studies that proposed user-centred interface design for digital libraries (Van House et al., 1996; Marchionini and Komlodi, 1998; Morgan, 1999; Baldonado, 2000; Meyyappan, Chowdhury & Foo, 2001; Meyyappan, Foo & Chowdhury, 2004). In this article, we present a structured, iterative methodology for user-centred design and evaluation of IR interfaces. Figure 1 illustrates our basic methodology. We recommend sequentially performing: (1) a competitive analysis, (2) user task analysis, (3) heuristic evaluation, (4) formative evaluation, and finally (5) a summative comparative evaluation. The methodology is explained first, and then we apply the methodology to iteratively design a prototype IR interface.

This paper used the Web of Science interface (available at http://wos.mimas.ac.uk) as a test-bed to conduct competitive analysis and user task analysis. This choice was made because the Web of Science is one of the best known and most widely used bibliographic services for the academic community in the UK. Since we used the Web of Science to perform competitive analysis, the resulting prototype would, therefore, be suitable for IR systems and environments like those of the Web of Science.

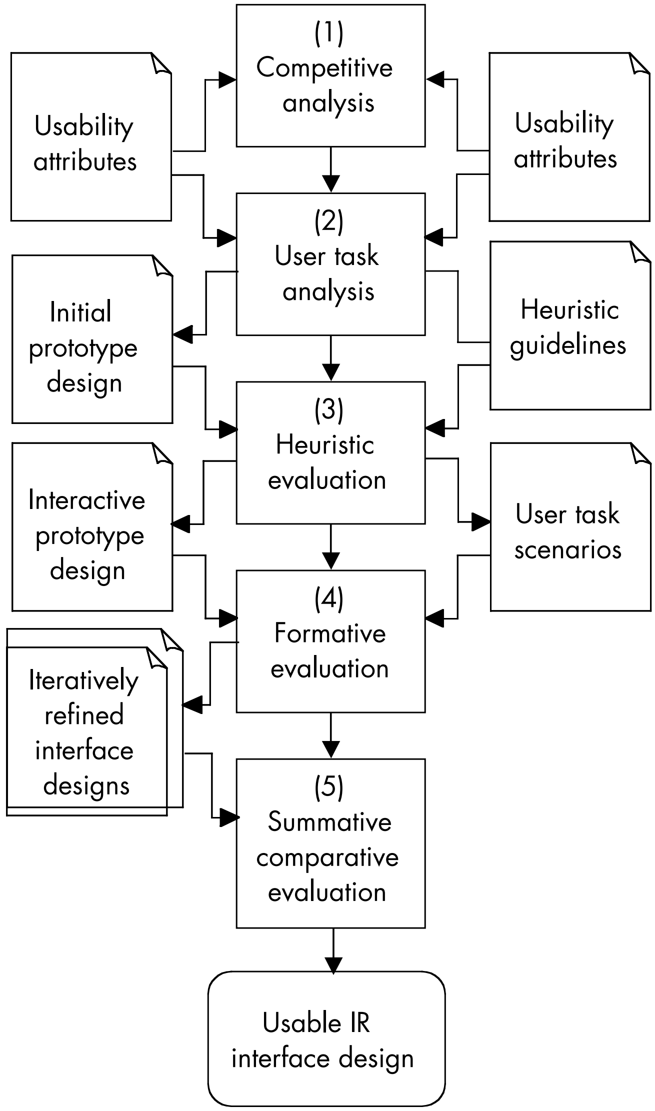

Figure 1. Methodology for user-centred design and evaluation of IR interfaces

2. METHODOLOGY

IR researchers interested in applying proven usability design and evaluation methods would discover few documented, well-tested methods for IR usability engineering. Our methodology, as illustrated in Figure 1, is based on sequentially performing:

1. a competitive analysis of an IR system to perform empirical usability testing;
2. a user task analysis based on user activities during usability tests;

3. an interactive prototype design drawn from task analysis;
4. heuristic evaluation of the prototype design using heuristic guidelines;
5. a Web-based prototype design, incorporating input from heuristic evaluation;
6. a formative evaluation of the Web-based prototype using task scenarios;
7. a revised prototype design based on formative evaluation;
8. a summative evaluation of the prototype design and a comparison of the results with competitive analysis for performing the same search tasks.

We describe each technique in more detail below. While similar methodologies have been applied to designing traditional GUI-based computer interfaces, this particular methodology is novel because we specifically designed it for, and applied it to, IR interfaces.

2.1 Performing competitive analysis

Performing a competitive analysis means analysing similar IR systems according to established usability guidelines. Usability testing with similar systems could help us to see how the functionality of the interface supports user tasks. The attributes that are frequently studied in usability testing include: learnability, efficiency, retention over time, error rate and subjective satisfaction (Shneiderman, 1998). Usability testing always involves real users as participants in the tests. The number of participants in a test depends on how many sub-groups should be covered, how much time and money could be used and how important it is to get statistically significant results. Studies have showed that the first four or five users reveal most problems and additional participants are likely to reveal few new major problems (Virzi, 1992; Nielsen, 1994). However, this sample size is too small to identify significant differences between groups. Spyridakis (1992) argued for a minimum of 10 to 12 participants.

Usability testing generally takes place in a specially equipped usability lab. Users are brought into the laboratory, where they perform a set of benchmark tasks. An effective technique during usability testing is to invite users to 'think-aloud' about what they are doing. User remarks obtained in usability testing could provide ideas for the new system design. Videotaping is often used for capturing users performing tasks for later review and to identify the problems that users encounter. Another useful technique is transaction logging, which unobtrusively creates a record of how a user performed a benchmark task. Questionnaires can be used to assess users' satisfaction with the interface. The Questionnaire for User Interface Satisfaction (QUIS) has been applied in a number of usability tests and has proved useful (Chin et al., 1988).

2.2 User task analysis

The competitive analysis would help us to see how users interact with the system and interfaces and should lead to a more efficient and effective interaction design. A user task analysis is important as an early input to the new interaction design. It provides a complete description of tasks, sub-tasks and the methods required to use a system in order to perform tasks. The task analysis could be carried out using the following stages:

1. Identify and group the tasks to be performed.
2. Break down the tasks from top to bottom showing detailed task descriptions, sequences and relationship amongst the tasks.
3. Record details of interactions between the user and the current system and any problems related to them.
4. Highlight areas where task processes are poorly understood, or are carried out differently by different users, or are inconsistent with the task structure.

A user task analysis provides the basis for new design and evaluation in terms of what types of tasks and task sequences users will need to perform within IR environments. Without a clear understanding of user task requirements, designers must guess the desired functionality of the new IR system, which would inevitably lead to poor interaction design. The closer the match between user task analysis and actual end-user tasks, the better and more effective the final user interaction design.

2.3 Heuristic evaluation

Heuristic evaluation is a usability inspection method. It is beneficial to carry out a heuristic evaluation on early prototypes before actual users are brought in to help with further testing. The results will generate good ideas for improving the interaction design. A number of studies showed that the design feedback provided by heuristic evaluation is valid and useful (Jeffries et al., 1991; Desurvire et al., 1992; Cuomo and Bowen 1994; Doubleday et al., 1997; Cogdill, 1999; Peng et al., 2004).

Heuristic evaluation always uses a short list of heuristic principles and three to five evaluators. Each evaluator is given a short list of heuristic principles to go through the interface independently to identify problems. They are not allowed to communicate with each other until all evaluations are completed. The theory behind independent evaluations is that a single evaluator would miss out many problems but different evaluators will find different problems. Thus, much better results can be obtained by combining the results from several evaluators. Nielsen (1992) recommended using 'double' usability experts who are specialists in both usability engineering and the user interface to be evaluated to ensure optimal results.

Nielsen's (1994) list of heuristic guidelines has been frequently used in heuristic evaluations. This list contains

the following 10 heuristics: visibility of system status; match between system and the real world; user control and freedom; consistency and standards; error prevention; recognition rather than recall; flexibility and efficiency of use; aesthetic and minimalist design; help users recognize, diagnose, and recover from errors; and help and documentation.

2.4 Formative evaluation

Formative evaluation ensures usability of interaction design by including users early in the design process. It aims to iteratively improve a new interaction design. Because formative evaluation involves real users, it uncovers usability problems that an expert performing heuristic evaluation might be unaware of.

Figure 2 shows the major stages involved in the formative evaluation. It begins with development of user task scenarios. Task scenarios derive from results of the user task analysis. Moreover, these scenarios should provide adequate coverage of all tasks as well as accurate flow of tasks identified during the user task analysis. Representative users perform these task scenarios and designers collect mostly qualitative data in the form of critical incidents. A critical incident is typically a problem encountered by a user such as an error, failure to complete a task scenario, or user confusion that affects the task flow or task performance. These data are analysed to identify user interaction components or features that both support and detract from user task performance. These observations are in turn used to suggest changes in the user interaction design.

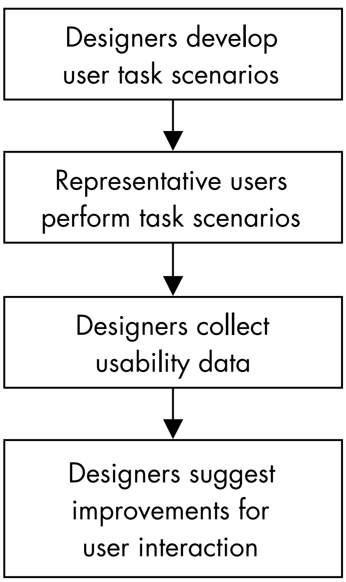

Figure 2. Stages in formative evaluations

2.5 Summative comparative evaluation

In contrast to formative evaluation, summative comparative evaluation is an empirical assessment of an interaction design in comparison with similar competitive systems for performing the same user tasks. Summative evaluation is typically performed with some more or less final version of the interface design to yield primarily quantitative data. The purpose of the summative comparative evaluation is to statistically compare different usability attributes with competitive systems. The same set of tasks that were used in analysing competitive systems could be compared in summative evaluation to see the design's ability to support user task performance. Similarly, users' satisfaction rating could be compared to see whether satisfaction with the new interface improved.

2.6 Prototyping and iterative design

We suggest early prototyping and an iterative approach to designing IR interfaces. Early prototyping could save time and cost and can be tested with real users. Based on the usability problems and opportunities disclosed by the testing, a new version of the interface can be created and tested. Prototypes are grouped into broad two categories: low-fidelity and high-fidelity (Rudd et al., 1996). A low-fidelity prototype could be as simple as a paper-and-pencil mock-up that shows general flow throughout the screens. High-fidelity prototypes, on the other hand, are typically built with software tools and can be programmed to simulate the functionality in the final product. Some studies compared low- and high-fidelity prototypes in identifying usability problems in a user interface (Nielsen, 1990; Virzi et al., 1996). A general requirement for the prototype, however, is that it can be developed quickly and modified with a minimum of effort.

Iterative design is based on a cyclic process of prototyping, testing and refining a user interaction design. A few studies showed that iterative design alone could improve the usability of a user interface (Bailey, 1993; Nielsen, 1993). In this research, we went through three major iterations during the prototype interface design, each consisting of the progression of usability methods described in our user-centred methodology.

3. APPLYING THE METHODOLOGY TO DESIGN A PROTOTYPE IR INTERFACE

We applied our user-centred methodology to iteratively design a prototype IR interface. As mentioned, we used the Web of Science to perform a competitive analysis. This allowed us to see how both novice and experienced searchers interacted with the Web of Science interface. Their performance of benchmark tasks produced task analysis and initial design of a prototype IR interface. We performed extensive evaluations of our prototype design. After each round of evaluation, the prototype was modified as needed within an iterative design process

described earlier. Finally, we performed a comparative evaluation of our prototype design with the Web of Science results.

3.1 Performing competitive analysis

The competitive analysis with the Web of Science was conducted in two separate but similar usability tests. The first test measured users' performance and satisfaction with the Web of Science (Ahmed et al., 2004). The second experiment examined novices' learning and retention with the interface (Ahmed et al., 2005). The first five out of the following seven tasks were obtained from a user survey at Loughborough University. Task 6 and Task 7 were taken from the Web of Science: Questions and Workbook (available at http://wos.mimas.ac.uk/documentation.html). All these tasks were used in both usability tests with the Web of Science.

Task 1: Find information on the topic of computer-aided design
Task 2: Find information about e-commerce
Task 3: Find information on concurrent engineering in construction
Task 4: Find information about applications of fibre optics
Task 5: Find works of Lawrence R. Rabiner
Task 6: Find work produced by the researchers in the Chemical Engineering department at UMIST
Task 7: Find articles citing work by M. Smith published in the journal of *Addictive Behaviors*

The usability tests took place in a specially equipped usability lab in the Department of Information Science at Loughborough University. A notice was placed in the Loughborough University website asking for volunteers for the usability tests. All interested participants were contacted through e-mails and asked about their IR experience. They were not told that the IR interface being evaluated was the Web of Science until they arrived for the tests.

The first usability test was carried out with two groups of volunteers recruited from various departments at Loughborough University. The novice group (n = 10) had no prior search experience. The experienced group (n = 12) were experienced in the search process and some had formal training in conducting online searching. Participants came one at a time for the test. At the beginning of each session, each participant was given a brief description of the purpose of the test and the experimental procedures of the session that would be followed. Novice searchers were given 15 minutes for free exploration of the Web of Science. For experienced users, this was not needed, since they were already familiar with the search process. All participants were then given the tasks and told to try to work on their own. We recorded each user search session through transaction logs and computer screen recordings using Lotus ScreenCam. After completion of all search tasks, participants completed a questionnaire on their satisfaction with the Web of Science interface.

The following performance variables were measured during our usability testing with the Web of Science interface:

1. Time taken: the total time taken to complete each task;
2. Search terms used: the number of different search terms used for each task;
3. Success score: successful completion of each task (1 = success, 0 = fail);
4. Error rates: number of different errors made for each task.

Table 1 shows users' performance results for each task in the first usability test.

Participants rated their subjective satisfaction with the Web of Science interface on a 7-point scale. Their satisfaction rating in the first test is shown Table 2.

The second test measured novices' learning and retention with the Web of Science. Ten naive searchers took part in the second test. None of them had taken part in our

Table 1. Users' performance data in first usability test (see Ahmed et al., 2004)*

Search tasks	Time taken (mins)		Search terms used		Success score		Error rates	
	Novice	Experienced	Novice	Experienced	Novice	Experienced	Novice	Experienced
Task 1	6.20 (4.02)	7.50 (4.62)	1.90 (1.37)	2.67 (1.67)	0.90 (0.32)	1.00 (0.00)	0.20 (0.63)	0.25 (0.45)
Task 2	4.50 (5.34)	4.33 (2.70)	2.30 (2.21)	1.58 (0.67)	1.00 (0.00)	1.00 (0.00)	0.10 (0.32)	0.00 (0.00)
Task 3	5.80 (6.16)	4.08 (2.67)	2.30 (1.34)	1.83 (1.27)	0.00 (0.00)	0.58 (0.51)	1.30 (0.67)	0.58 (1.16)
Task 4	4.90 (3.41)	4.33 (3.05)	2.80 (1.69)	3.25 (2.09)	0.20 (0.42)	0.67 (0.49)	0.90 (0.32)	0.83 (0.94)
Task 5	5.60 (3.37)	4.00 (2.17)	4.00 (1.49)	1.75 (1.06)	0.60 (0.52)	0.92 (0.29)	3.10 (1.73)	0.50 (0.80)
Task 6	8.30 (4.24)	10.5 (4.54)	3.40 (1.26)	4.33 (2.46)	0.10 (0.32)	0.17 (0.39)	2.30 (1.57)	2.33 (0.98)
Task 7	8.60 (5.64)	8.08 (4.60)	3.20 (1.87)	3.83 (1.64)	0.60 (0.52)	0.83 (0.39)	2.10 (2.33)	1.17 (1.03)
Overall	43.90 (17.63)	42.83 (11.79)	19.80 (5.87)	20.00 (5.61)	3.40 (1.51)	5.17 (1.33)	10.00 (4.29)	5.67 (2.19)

* Means (and standard deviations)

Table 2. Users' satisfaction rating in first test (see Ahmed et al., 2004)

Question	Novice	Experienced	Question	Novice	Experienced
Overall reactions			**Terminology and system feedback**		
Terrible vs. wonderful	4.70 (0.82)	4.42 (0.79)	Simple and natural dialogue	4.80 (2.20)	4.25 (1.06)
Unimpressive vs. impressive	4.90 (0.99)	4.42 (1.00)	Terms used in the system	5.10 (1.60)	4.58 (1.62)
Difficult vs. easy	4.70 (1.64)	4.17 (1.34)	Position of message	5.40 (1.58)	4.75 (1.14)
Inefficient vs. efficient	4.70 (1.42)	4.42 (1.56)	Prompts for input	4.30 (2.16)	4.08 (1.31)
Useless vs. useful	5.40 (1.58)	5.58 (0.90)	Inform about work progress	4.20 (1.75)	4.50 (1.38)
Unfriendly vs. friendly	4.60 (1.78)	4.17 (1.27)	Error messages	4.11 (2.09)	3.30 (1.49)
Frustrating vs. satisfying	4.90 (1.60)	4.33 (1.30)	**Learning**		
Ineffective vs. powerful	4.89 (1.54)	5.25 (1.14)	System learning	5.60 (1.35)	4.67 (1.30)
Dull vs. stimulating	4.70 (1.34)	3.92 (1.38)	Exploring by trial and error	5.40 (1.35)	5.42 (1.31)
Rigid vs. flexible	4.56 (1.51)	4.09 (1.22)	Remembering commands	5.33 (1.32)	4.64 (1.57)
Screen			Performing tasks is simple	4.67 (1.66)	4.42 (1.08)
Reading characters	5.60 (1.84)	5.83 (1.03)	Help messages on the screen	4.89 (1.45)	4.73 (1.10)
Onscreen information	4.50 (1.35)	4.25 (1.36)	Help access	5.11 (2.15)	5.00 (0.94)
Information arrangement	5.20 (1.32)	4.17 (1.34)	**System capabilities**		
Easy to find information	4.60 (1.65)	3.92 (1.24)	System speed	4.80 (1.75)	4.83 (1.59)
Screen sequencing	4.70 (1.89)	4.33 (1.30)	System reliability	5.11 (1.62)	5.27 (1.19)
Screen back track	4.60 (1.58)	4.50 (1.62)	Correcting mistakes	4.80 (1.14)	5.50 (1.17)
Back to main screen	5.40 (1.78)	5.33 (1.56)	Designed for all levels of users	4.44 (1.67)	4.25 (1.48)

first test. The equipment and the search tasks used in the second test were the same. The experimental procedure was also identical, except that all naive users participated in two sessions spaced four weeks apart. The first session measured novices' learning with the Web of Science interface. The second session measured their retention of search skills after not having used it for four weeks.

Table 3 shows novices' performance data in the second usability test. Their subjective satisfaction in both learning and retention is shown in Table 4.

3.2 User task analysis

These initial experiments resulted in task analysis and the initial requirements for our prototype interface. Important IR-related tasks identified included both simple tasks such as search for an author/person, or a single word or a phrase and advanced tasks that required the use of Boolean operations, truncation, proximity etc. We found that while skills necessary for performing simple tasks were readily available to most users, considerably more experience was required for advanced searching. Based on this, our prototype design offered both a 'simple' search option for simple straightforward queries and a more elaborate 'advanced' one for complex queries. The simple search interface would provide only a minimal subset of commands and actions. This would help novices to learn to use the system quickly. Moreover, they are most likely to make correct choices when only a few options are shown. Novices would venture on to more complex searching as they became familiar with the interface.

Table 3. Novices' performance data in the second test (see Ahmed et al., 2005)

Search tasks	Time taken (mins)		Search terms used		Success score		Error rates	
	Learning	Retention	Learning	Retention	Learning	Retention	Learning	Retention
Task 1	2.90 (1.29)	2.90 (1.85)	1.30 (0.48)	1.30 (0.95)	1.00 (0.00)	1.00 (0.00)	0.20 (0.42)	0.20 (0.63)
Task 2	1.70 (0.95)	1.60 (0.84)	1.00 (0.00)	1.20 (0.42)	1.00 (0.00)	0.90 (0.32)	0.00 (0.00)	0.30 (0.48)
Task 3	2.50 (1.35)	2.00 (0.94)	1.30 (0.48)	1.50 (0.53)	0.60 (0.52)	0.50 (0.53)	0.60 (0.70)	0.80 (0.63)
Task 4	2.20 (1.14)	2.00 (1.05)	1.70 (1.25)	2.00 (1.15)	0.60 (0.52)	0.60 (0.52)	0.60 (0.70)	0.70 (0.67)
Task 5	2.50 (1.58)	1.80 (0.92)	1.80 (1.14)	1.40 (0.70)	0.80 (0.42)	0.70 (0.48)	0.90 (0.99)	0.70 (0.82)
Task 6	7.30 (4.57)	7.80 (4.39)	3.20 (1.14)	2.80 (1.55)	0.60 (0.52)	0.20 (0.42)	2.00 (1.33)	1.50 (0.97)
Task 7	4.70 (3.37)	3.50 (0.97)	2.20 (2.10)	1.20 (0.42)	0.80 (0.42)	0.70 (0.48)	0.70 (1.89)	0.30 (0.48)
Overall	23.80 (9.463)	21.60 (6.62)	12.50 (3.66)	11.40 (1.71)	5.40 (2.07)	4.60 (1.78)	5.20 (3.97)	4.50 (1.51)

Table 4. Novices' satisfaction rating in second test (see Ahmed et al., 2005)

Question	Learning	Retention	Question	Learning	Retention
Overall reactions			**Terminology and system feedback**		
Terrible vs. wonderful	5.30 (0.48)	5.20 (0.92)	Simple and natural dialogue	4.90 (0.88)	4.90 (1.20)
Unimpressive vs. impressive	4.90 (1.10)	5.10 (0.88)	Terms used in the system	5.10 (0.57)	5.20 (1.14)
Difficult vs. easy	5.40 (1.26)	5.10 (1.60)	Position of message	5.20 (0.92)	5.50 (0.71)
Inefficient vs. efficient	4.90 (0.99)	5.20 (1.48)	Prompts for input	4.70 (1.83)	4.90 (1.29)
Useless vs. useful	5.90 (1.10)	5.50 (1.08)	Inform about work progress	4.70 (1.57)	5.20 (1.03)
Unfriendly vs. friendly	5.30 (1.49)	4.50 (1.72)	Error messages	3.50 (1.72)	4.10 (0.74)
Frustrating vs. satisfying	5.60 (1.17)	4.80 (1.93)	**Learning**		
Ineffective vs. powerful	5.30 (0.95)	5.30 (1.05)	System learning	5.60 (1.51)	5.10 (1.91)
Dull vs. stimulating	4.40 (0.97)	4.50 (1.65)	Exploring by trial and error	4.60 (1.51)	4.30 (1.83)
Rigid vs. flexible	4.70 (1.34)	4.60 (1.58)	Remembering commands	4.90 (1.20)	4.20 (1.87)
Screen		4.90 (1.60)	Performing tasks is simple	5.40 (1.51)	4.90 (0.88)
Reading characters	5.90 (0.88)	5.20 (1.03)	Help messages on the screen	4.90 (1.29)	4.30 (0.95)
Onscreen information	5.00 (1.49)	5.00 (1.05)	Help access	4.44 (0.88)	4.40 (1.35)
Information arrangement	5.60 (0.84)		**System capabilities**		
Easy to find information	5.00 (0.82)	4.78 (1.48)	System speed	3.80 (1.75)	4.50 (1.58)
Screen sequencing	5.60 (0.84)	4.60 (1.35)	System reliability	4.90 (1.73)	4.10 (1.79)
Screen back track	4.20 (1.81)	3.90 (1.91)	Correcting mistakes	4.40 (0.84)	4.10 (1.45)
Back to main screen	5.33 (1.41)	5.89 (1.05)	Designed for all levels of users	4.30 (1.34)	4.20 (2.04)

We saw most users did not take advantage of the more sophisticated capabilities like Boolean searches, truncation and proximity, and instead performed simple searches. Some users only used Boolean ANDs rather than developing more complex queries using OR relationships. Overall, users have a poor understanding of search strategy and lack suitable mental models as evidenced by the users who enter actual sentences to query the system. There is no doubt that Boolean query formulation is difficult for users. One possible solution to this problem can be giving up Boolean search entirely. Some studies, however, showed that Boolean searching is no worse than any other known approach (e.g. Frants et al., 1999). Visual query formulations may be an important step towards improving the Boolean searches. Boolean operators could be presented into the interface to users as options to select in a template. This would help naive users to pose complex queries and aid their retention of search skills over time. Therefore, we chose visual query formulations as a major focus in our prototype design.

Only a few users tried alternative terms in their queries. We believe that helping users in query articulation could lead to improved search performance. Thus, we decided to include an index browsing feature into our prototype design. This would help users overcome the problem of complex query formulations. Moreover, at the time this research was carried out, the Web of Science interface did not offer search histories. The lack of search histories certainly affected naive users who were not able to see their search pattern developing into a complex search. Thus, we decided to add search histories into our prototype design to help users combine their previous search sets and to diagnosis inappropriate searches. Also, relevance feedback was not well supported. The link to 'find related records' appeared only in full record displays. IR interfaces need to actively encourage users to use such features by providing more visible links from result pages.

Finally, users had real difficulties in using abbreviations for source titles in the Web of Science. This list was very long, and they had to copy and paste (or type) the name from the list into the source text box. The address field was also heavily abbreviated. The help screens gave a long list of common abbreviations to help find what users wanted. IR interfaces should really make these features easier to handle, perhaps avoiding using abbreviations altogether. While spreading information over several screens may be graphically appealing, the burden of navigating from one search screen to another was evident during usability tests. Compact design might take a longer time to scan, but it would take much less time than scanning several screens. Thus, we decided to adopt a compact screen design for the prototype which minimized scrolling and jumping and anchored users in a screen space that tightly couples search and result pages.

3.3 Designing the prototype IR interface

Based on user task analysis, the initial design of a prototype interface was proposed. The design was first sketched on paper and then mocked-up in Microsoft

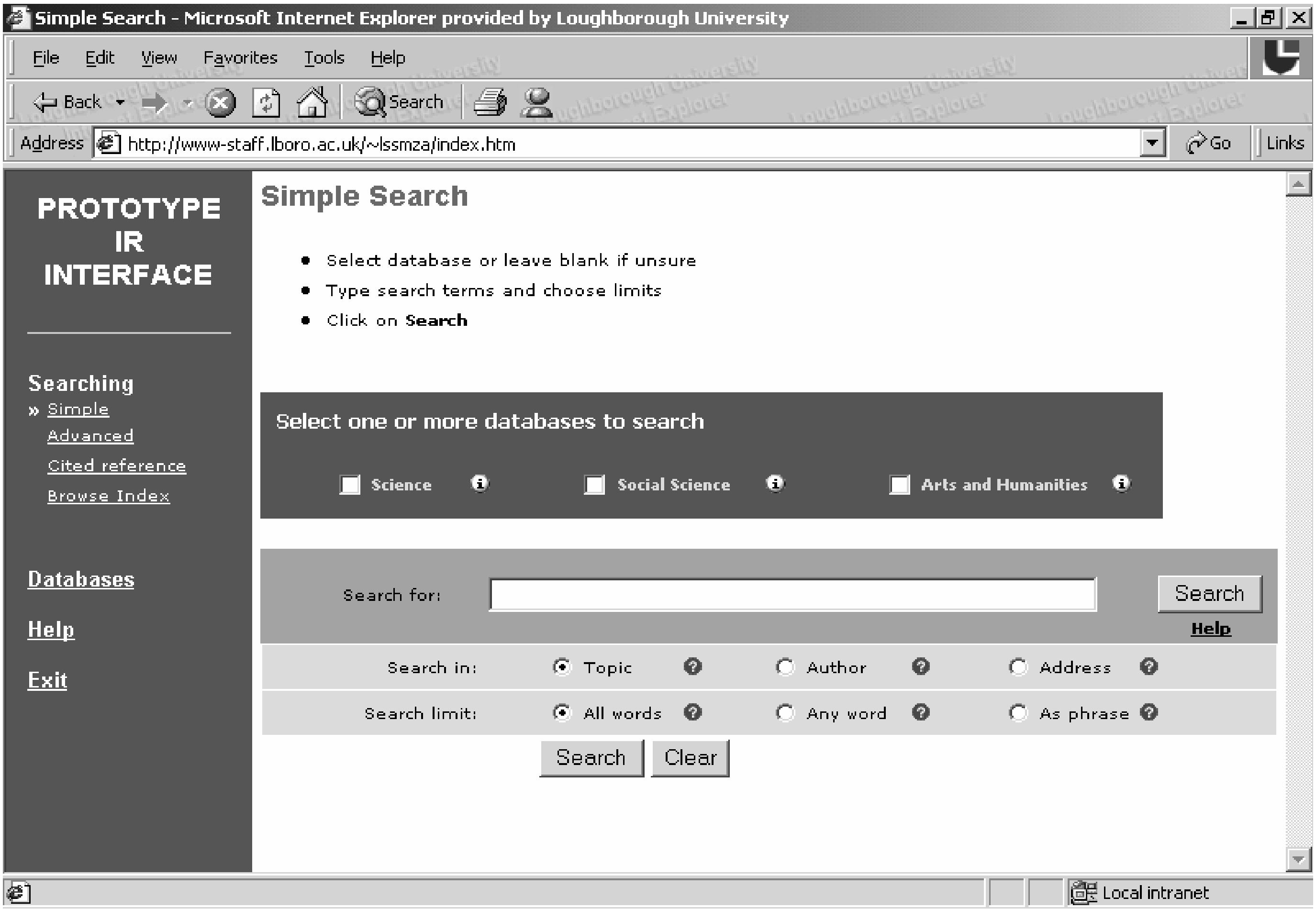

Figure 3. Simple search screen

PowerPoint using Visual Basic objects. The initial screen opens with a simple search interface for straightforward queries. The simple search enables users to choose from a selection of common simple ways to combine Boolean searches into the queries. From the simple search screen, the users can begin a search, or jump directly to the advanced search or cited reference search or to browse index. The navigation menu at the left provides basic navigation within the site. Figure 3 shows our simple search screen.

Advanced search offers text boxes where the searcher can enter words for topic, author, source title and address search. In addition, the searcher can limit the number of retrieved documents to be displayed per result page. Figure 4 shows our advanced search screen.

The result screen shows the search results for a matched query. The query statement appears in a text box which allows users to enter a complete search query including Boolean and proximity operators, and appropriate field labels (Topic=) for efficient searching. Figure 5 illustrates our results screen.

The previous searches are available from both the search screens and the result screen. Users can use previous searches to recall previously searched items and to further refine the search query. The results are displayed 10 results at a time for simple search or as defined by the users in advanced search. The titles are hyperlinked to the full records. 'Show similar records' link retrieves other articles that cite at least one of the same references. The users can mark individual records for printing or downloading. The 'Mark all' selects all records retrieved during a search.

3.4 Prototype heuristic evaluation

During our heuristic evaluation, three expert members carried out their independent evaluation using a

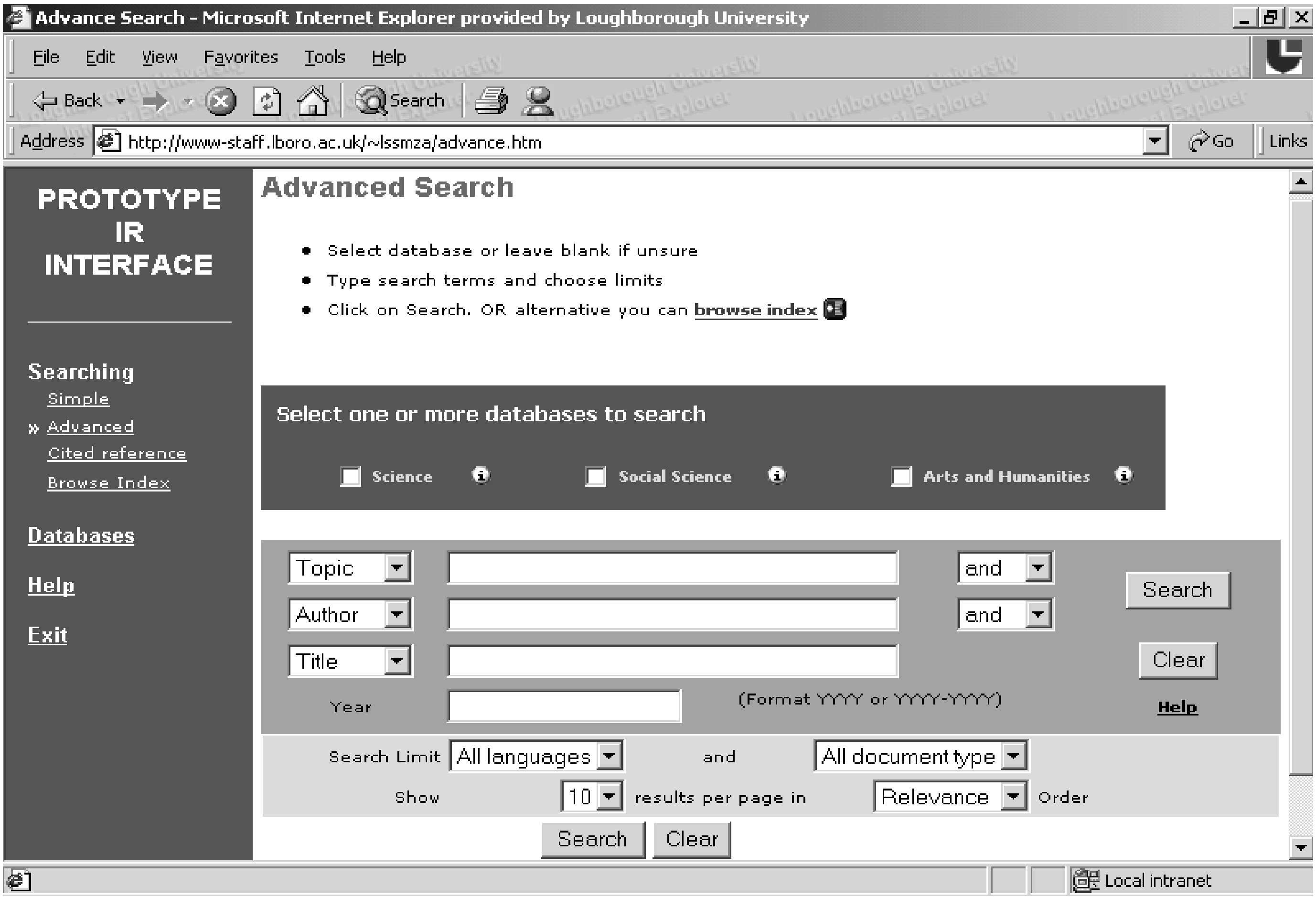

Figure 4. Advanced search screen

PowerPoint version of the prototype design. The experts were Martin Maguire, Anne Morris and John Richardson. Maguire and Richardson were affiliated with the Human Sciences and Advanced Technology (HUSAT) Research Institute at Loughborough University. Morris was a Reader in the Department of Information Science at Loughborough University. They were all experienced in evaluating a wide variety of systems and interfaces. They conducted their independent evaluation first and then we combined and analysed the findings according to Nielsen's heuristics.

3.4.1 Findings

Visibility of system status: The prototype was generally well signposted, with titles on all screens. The highlighting of the current option selected on the navigation bar was also helpful. In addition, radio button selection indicated the choices made.

Match between system and the real world: Although the language used was simple and straightforward, experts suggested that some terms such as 'cited reference' and 'index' needed to be explained within help. They commented that the 'search limits' instructions were too concise, in other words, the phrase 'all words' could apply to the search term or retrieved records. The experts recommended explaining these options by using '?' keys on simple search interface.

User control and freedom: The prototype presented a good range of search functions and options in a simple way, which most users should be able to control fairly easily. The options were available on the navigation bar so searchers knew where they were at all times. On advanced search, the drop-down menus provided good flexibility and the different default (topic, author, source, address etc.) was a good way to highlight the options available. The search limits were also useful.

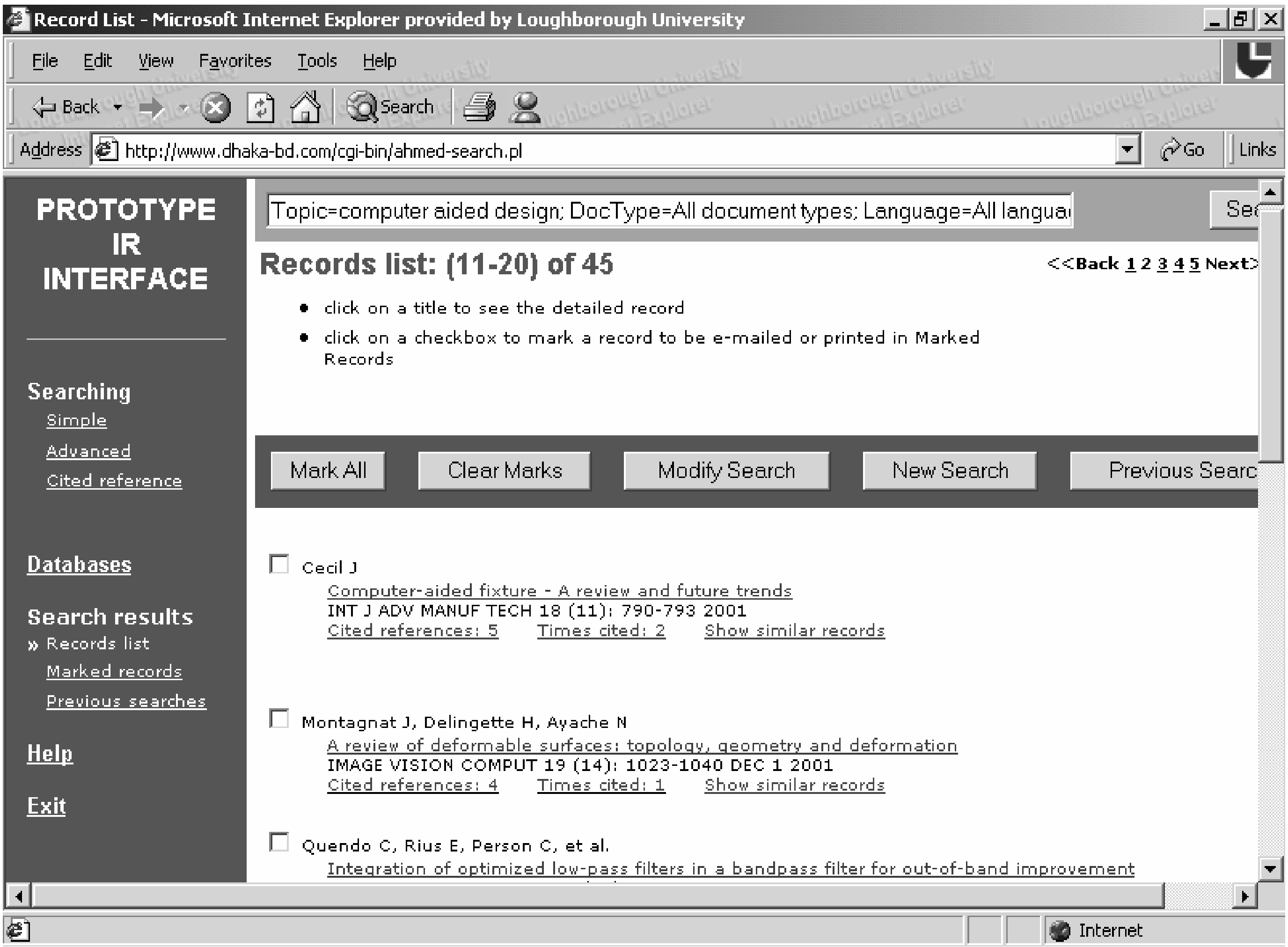

Figure 5. Search results screen

Consistency and standards: The prototype presented a fixed screen format which provided the user with a consistent look. All options were presented either as a button or underlined link throughout. However, they noted that there were a few instances where the interface consistency could be enhanced. The difference between the 'help' and '?' was not clear. Also, there was a 'help' link on simple search in the main working area but not on other screens. Experts suggested lining up the check boxes on the simple search and search limit options on the advanced search screens. Search and Clear buttons should be aligned on each screen.

Error prevention: The prototype was based on a simple 'point and click' interaction which would help reduce errors. However, a means of detecting possible misspellings in search terms or author names could be useful.

Recognition rather than recall: The prototype assisted the users in recognition rather than having to recall commands. The user was required to respond to prompts not generate search terms and fields. Information was clearly visible and help was readily available. For radio buttons, however, experts suggested that the interface select one option by default to show users what to do and to indicate the most common option.

Flexibility and efficiency of use: The prototype provided a flexible and efficient interface where a search could be performed with just a few user actions. The user could move flexibly between options using the navigation bar. For a long list of hits, however, the user might

not realize that the 'mark all' and 'clear all' buttons were present at the bottom of the results page. Experts suggested that these buttons should also appear at the top of the results page.

Aesthetic and minimalist design: The design was admirably concise – only the critical options were included and the labelling and layout were clear. Redundant coding (colour and grouping) were used to good effect. It also worked in greyscale, which would help users with restricted colour vision.

Help user recognize, diagnose, and recover from errors: The prototype helped the user to undo actions and to recover from errors. They noted that a help option explaining the various types of link on the search records screen could be useful.

Help and documentation: The prototype interface provided readily accessible help (both general and contextual) which should help users reduce errors that might occur.

Based on the findings of heuristic evaluations, a Web-based version of the prototype interface was developed. The Web-based prototype remedied many of the design flaws revealed in the heuristic evaluation.

3.5 Formative evaluation

We carefully created four user task scenarios. We thoroughly pre-tested and debugged all scenarios before presenting them to volunteers during formative evaluations. Six volunteers took part in the formative evaluations. They were all postgraduate students at Loughborough University. During each of six formative evaluation sessions, a formal protocol was followed. This consisted of welcoming the user, giving them an overview of the evaluation to be performed, and then explaining the prototype. They were then allowed to play with the prototype for about 15 minutes. This was done to help users familiarize themselves with the interface. Once a user had successfully figured out how to use the prototype interface, they were asked to perform the task scenarios. Users were observed while performing the scenarios and their interaction was recorded by Lotus ScreenCam. Their critical incidents were noted for both positive and negative reactions to the prototype design. Interviews were conducted with each participant following each observation to extract additional information and feedback.

Since the evaluation was not meant to be statistically analysed, the observations made from the formative evaluations could only be explained qualitatively. From a usability standpoint, the evaluation results were quite satisfactory. All volunteers completed the task scenarios. Much of the comments obtained in the formative evaluations were also positive and complimentary toward the prototype design. Though the reactions were gratifying, they were not overly useful for making revisions to the prototype interface. However, a few of the user comments did help identify areas needing improvements. For example, two volunteers commented that the feedback given on retrieval of a null set was basic. They suggested that the interface could indicate methods of diagnosing search failure and provide hints for more successful strategies. Some volunteers also noted that the font size used in the prototype was too small.

After formative evaluations, a few minor changes were made to the prototype design. A search error screen was added (see Figure 6) which appears when the user types in a query term which does not match with any items in the prototype. It shows a list of search tips for effective query formulations. At the bottom of the error screen is a link back to the search page to modify the query. Font sizes were enlarged to help those users who have limited vision.

The volunteers in formative evaluations expressed an overall positive reaction to the colour scheme. Since most of the users felt the use of colours were appropriate, the colour scheme was left unchanged.

3.6 Summative comparative evaluation

As with usability tests during competitive analysis, all interactions with the Web-based prototype interface were recorded and analysed. To compare the results of prototype evaluation with earlier tests, sample results pages for the prototype were generated in HTML for task-specific queries used in the usability tests. Two different search groups took part in the summative evaluation. None of them participated in our earlier tests with the Web of Science. The novice group (n = 10) included three postgraduates, six research students and one member of research staff. They all reported not having used IR systems prior to this evaluation. There were four male and six female users in this group. The experienced group

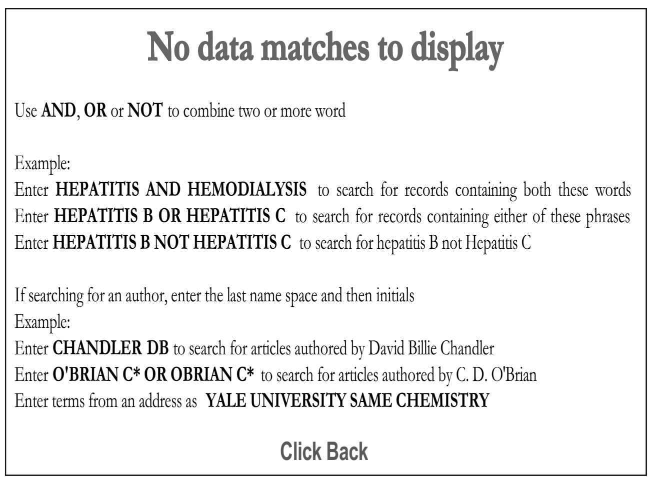

Figure 6. Search error screen

($n = 10$) consisted of four postgraduates, four research students and two research staff. There were six male and four female users in this group. They all had reported having varying levels of experience in using IR systems. Three experienced searchers reported having had formal training on using IR systems.

The software and equipment used in this experiment were the same as those used in our earlier usability tests. The experimental procedure was also identical except that the users were not allowed to play with the prototype before the evaluation. Similarly, the same set of search tasks as in Section 3.1 was used. The questionnaire used to collect users' subjective satisfaction with the prototype interface was also similar. Table 5 shows users' performance data with the prototype interface.

The independent sample *t*-test showed that there was no significant difference in time taken, terms used or error rates between novice and experienced searchers. However, the results of success score showed a significant difference ($p < 0.05$) in performance. Since experienced users had experience in using IR systems, it is quite reasonable that they would perform better than the naive users. The ANOVA results for performance among different age, gender, training, computer experience and status groups showed no significant difference among groups.

The data collected through the QUIS are summarized in Table 6. The Mann-Whitney *U*-test results showed no significant difference in subjective satisfaction by novice and experienced searchers with the prototype interface.

We performed a comparative analysis of our prototype evaluation data with earlier tests. An independent sample *t*-test was run to see the difference between experienced users in the first usability test and the prototype summative evaluation. The result showed that experienced users in prototype evaluation performed significantly better ($p < 0.01$) than the first usability test in all performance measures. Similarly, ANOVA results for novices' performance across all experiments showed significant differences ($p < 0.05$) in all performance measures.

The Mann-Whitney *U*-test was used to compare the subjective satisfaction rating among different groups across all experiments. The results showed both novice and experienced searchers in the prototype evaluation were significantly more satisfied ($p < 0.05$) with the interface than their counterparts in the first usability test. Novices' subjective satisfaction in the retention session and the prototype evaluation was also significant ($p < 0.01$), although no significant difference was found between novices' satisfaction in the learning session and the prototype evaluation.

4. DISCUSSION AND CONCLUSIONS

The summative comparative evaluation examined users' performance and satisfaction while interacting with the prototype IR interface in searching for task-specific queries. The comparative analysis of the results shows that the prototype interface enabled both novice and experienced users to improve their performance significantly compared to earlier usability tests with the Web of Science. Whereas searchers in the competitive analysis took more time to complete the tasks, the users of the prototype evaluation took much less time. They were more successful and made fewer errors compared to earlier usability tests. The prototype also diminished the individual differences among different gender, age, computer experience and status groups. Also, both novice and experienced users in the prototype evaluation were more satisfied with the prototype interface than with the Web of Science. Given that the search tasks were identical, any difference in performance and satisfaction observed should be due to the quality of the interface and different levels of users' search experience. These results indicate the efficacy of our user-centred methodology.

Based on our experience with the prototype design, we offer designers a few general principles for designing

Table 5. Users' performance data with the prototype interface

Search tasks	Time taken (mins)		Search terms used		Success score		Error rates	
	Novice	Experienced	Novice	Experienced	Novice	Experienced	Noviceg	Experienced
Task 1	2.00 (0.82)	2.10 (0.74)	1.10 (0.32)	1.30 (0.67)	1.00 (0.00)	1.00 (0.00)	0.10 (0.32)	0.00 (0.00)
Task 2	1.50 (0.71)	1.80 (1.03)	1.10 (0.32)	1.30 (0.48)	1.00 (0.00)	1.00 (0.00)	0.10 (0.32)	0.00 (0.00)
Task 3	2.30 (0.82)	2.10 (0.57)	1.40 (0.70)	1.10 (0.32)	1.00 (0.00)	1.00 (0.00)	0.10 (0.32)	0.00 (0.00)
Task 4	3.10 (1.45)	2.10 (0.57)	1.60 (0.97)	1.40 (0.70)	0.90 (0.32)	0.90 (0.32)	0.10 (0.32)	0.00 (0.00)
Task 5	3.40 (2.32)	2.10 (0.74)	2.20 (0.97)	1.80 (0.63)	0.70 (0.48)	0.90 (0.32)	1.10 (0.99)	0.80 (0.42)
Task 6	3.90 (3.03)	4.30 (1.64)	1.50 (0.53)	2.10 (0.99)	0.50 (0.53)	0.80 (0.42)	0.40 (0.70)	0.80 (0.92)
Task 7	3.70 (2.58)	2.10 (0.57)	1.30 (0.67)	1.00 (0.00)	0.60 (0.52)	1.00 (0.00)	0.20 (0.42)	0.30 (0.67)
Overall	19.90 (6.59)	16.60 (3.37)	10.20 (1.48)	10.00 (1.76)	5.70 (1.06)	6.60 (0.52)	2.10 (0.99)	1.90 (0.88)

Table 6. Subjective satisfaction with the prototype interface

Question	Novice	Experienced	Question	Novice	Experienced
Overall reactions			**Terminology and system feedback**		
Terrible vs. wonderful	5.20 (1.03)	5.00 (0.67)	Simple and natural dialogue	4.80 (1.62)	5.20 (1.32)
Unimpressive vs. impressive	5.10 (0.74)	4.90 (0.88)	Terms used in the system	5.33 (1.41)	5.60 (1.17)
Difficult vs. easy	5.10 (1.85)	5.40 (1.26)	Position of message	5.56 (1.59)	5.80 (1.48)
Inefficient vs. efficient	5.00 (1.15)	5.20 (1.55)	Prompts for input	5.30 (1.25)	5.00 (2.21)
Useless vs. useful	5.10 (1.52)	5.70 (1.06)	Inform about work progress	4.67 (1.66)	4.13 (2.42)
Unfriendly vs. friendly	5.20 (0.63)	5.40 (1.43)	Error messages	4.25 (2.05)	4.67 (2.12)
Frustrating vs. satisfying	4.50 (1.18)	5.00 (1.49)	**Learning**		
Ineffective vs. powerful	5.00 (0.67)	5.10 (0.99)	System learning	6.00 (1.56)	5.90 (1.60)
Dull vs. stimulating	4.70 (0.95)	4.90 (0.74)	Exploring by trial and error	5.40 (1.51)	5.80 (1.55)
Rigid vs. flexible	4.80 (1.32)	5.20 (0.79)	Remembering commands	5.90 (1.37)	5.50 (1.35)
Screen			Performing tasks is simple	5.70 (1.64)	5.70 (1.57)
Reading characters	5.00 (1.94)	4.60 (1.43)	Help messages on the screen	4.75 (2.19)	5.25 (1.17)
Onscreen information	5.20 (1.48)	4.80 (1.99)	Help access	5.50 (2.00)	4.67 (2.16)
Information arrangement	5.40 (1.35)	5.50 (1.65)	**System capabilities**		
Easy to find information	4.80 (1.40)	5.20 (0.92)	System speed	5.00 (1.33)	6.00 (1.22)
Screen sequencing	5.00 (1.76)	5.50 (1.35)	System reliability	5.11 (1.27)	4.67 (2.16)
Screen back track	4.70 (2.11)	5.30 (1.95)	Correcting mistakes	5.38 (1.60)	5.40 (1.58)
Back to main screen	5.60 (1.90)	5.70 (2.26)	Designed for all levels of users	5.20 (1.14)	5.10 (1.60)

effective IR interfaces. These guidelines are not intended to be an exhaustive list of principles to follow when designing interfaces for IR systems. Rather, they highlight some high-level concepts that should be considered in order to promote usability of IR interfaces. Some of these principles were followed in our prototype interface design and resulted in improved performance and satisfaction both by novice and experienced users.

1. *Strive for consistency*: Inconsistencies in search interfaces could affect searchers' performance and satisfaction significantly. It is important to make sure that layout, terminology, instructions, colour and fonts are used consistently across interfaces. User interfaces for IR applications also need to be consistent among themselves. Consistency across multiple interfaces could bring faster learning and retention, leading to more effective searches and higher satisfaction.
2. *Support both novice and experienced users*: IR interfaces need to support both novice and experienced users. Recognizing that not all users have the same level of proficiency, designers should consider developing a set of search screens to support their unique needs. For example, novices could be presented with a simple interface that provides the basic functionality of the search system. Alternative interfaces could be offered for more experienced users giving them more control, more options and more features. Expert users should be allowed to use shortcuts for entering commands rapidly and directly, perhaps even bypassing much of the feedback and error checking that should be supplied to novice users. The interface should also provide intuitive bridges between different search and result screens.
3. *Make the interface actions visible to the users*: The query formulation process needs to be visible to users. Displaying terms included in a query is a common standard for results lists. The results should also indicate whether or how Boolean operators have been used.
4. *Assist users in refining the search query*: The designers could employ a number of techniques. First, they should provide clear and simple ways to modify a query, and the ability to reset the query statement. Second, the query can be summarized on the results page so that users can be reminded exactly what was searched for. Once a search is made, the query terms can be displayed in a window on the top of the results list. This would allow the user to modify the query terms, thus reducing the need to navigate to search screens for query (re)formulation.
5. *Offer informative feedback*: The users should be informed about all aspects of the search process, such as the database(s), fields, what is being searched for etc. When the search is complete, it should be obvious to the user what happened and why. Results lists should be structured in such a way that the contents of the returned records are clear. The information most important to users should be clearly displayed in the results. The inclusion of the query

statement in the result and the highlighting of search terms are essential. These features help users to focus on their searches at all stages of the search process. They guide the users not only in selecting which records might be of interest, but also whether the search resulted in the types of information desired.

6. *Offer simple error handling*: All error messages should be specific, constructive, uncritical of the user, and should offer no more technical detail than necessary. The error messages should also indicate methods of diagnosing search failures and provide hints on more successful strategies. Users will find ways to make errors. Therefore, in order to increase the user's feeling of being in control, the interface should always offer an easy way out from the system. This would encourage users to perform exploratory learning since they could always try out unknown options, knowing they have the ability to get out of trouble without repercussions.
7. *Permit easy reversal of actions*: Every action should be reversible so users can go back to a previous state in a session. The users should be able to keep a history of queries and to return to search sets without re-keying the search query. This would give users the flexibility to refine searches as they gain greater understanding of the topic being researched.
8. *Avoid complex navigation*: Reduce the amount of required navigation by making the functions available on the search screen. The navigation buttons should always be present while the users build a query. A navigation menu containing the different search options should always be available while the users build the query.
9. *Reduce short-term memory load*: Designers could employ a number of techniques to reduce short-term memory load. A compact design that minimizes scrolling and jumping and anchors users in a screen space that tightly couples search and result is useful. Displaying Boolean operators as an option will allow users to select items through recognition rather than recall. The interface should describe the required format whenever users are asked to provide input and, if possible, provide an example of legal input.

There were several limitations with our prototype interface. It was relatively simpler in terms of design complexity when compared to any existing Web-based IR systems. In addition, it was implemented with limited functionality, although basic information for the tasks was included. This may occasionally have caused users to ask for fringe information which was not always available. In particular, the users were not able to use the index browsing facility. Instead, they added relevant query terms from their own knowledge of the information problems. This must have affected their choice of search terms. Furthermore, the prototype only allowed searching task-specific queries used in earlier tests. It displayed a default result page if a user's query was accurate. The users were not able to interact with the result pages, nor could they observe the context in which the results list appeared. To overcome these problems, the prototype would have to be fully functional and implemented in a real environment. However, the results of summative comparative evaluation showed significant improvements in users' performance and satisfaction with the prototype IR interface. This suggests that our user-centred methodology could yield better interface design for IR systems.

This paper has demonstrated that the user-centred methodology could have a major impact on designing IR interfaces. IR interface designs need to be user centred to support users' needs and their information-seeking behaviour. Until designers apply such techniques, most IR interface designs will be driven by the constructional domain, and possibly by computer scientists, rather than by the needs of the users for whom these systems are intended. We hope this work provides a starting point for techniques that let IR researchers and practitioners design IR interfaces that are both easy to learn, use and remember.

ACKNOWLEDGEMENTS

We would like to thank the Commonwealth Scholarship Commission in the UK for its financial support to one of us (SMZA). The contributions of expert panel members in heuristic evaluation and of the volunteer searchers are gratefully acknowledged. We also thank MIMAS for copies of the transaction logs.

REFERENCES

Ahmed, S.M.Z., McKnight, C. and Oppenheim, C. (2004) 'A Study of Users' Performance and Satisfaction with the Web of Science IR Interface', *Journal of Information Science* 30 (5): 459–68.

Ahmed, S.M.Z., McKnight, C. and Oppenheim, C. (2005) 'A Study of Learning and Retention with a Web-based IR Interface', *Journal of Librarianship and Information Science* 37 (1): 7–16.

Bailey, G. (1993) 'Iterative Methodology and Designer Training in Human-computer Interface Design', in *INTERCHI '93: Conference Proceedings in Human Factors in Computing Systems*, 24–29 April, Amsterdam, The Netherlands, pp. 198–204. New York: ACM.

Baldonado M.Q.W. (2000) 'A User-centered Interface for Information Exploration in a Heterogeneous Digital Library', *Journal of the American Society for Information Science* 51 (3): 297–310.

Barker, A.L. (1997) 'DataStar Web: Living up to the Hype? An Evaluation of the Interface and Search System', in *Online Information 97: Proceedings of the 21st International Online Information Meeting*, 9–11 December, London, pp. 213–22. Oxford: Learned Information.

Chin, J.P., Diehl, V.A. and Norman, K.L. (1988) 'Development of an Instrument Measuring User Satisfaction of the Human-computer Interface', in *CHI '88: Proceedings of the Conference on Human Factors in Computing Systems*, 15–19 May, Washington, DC, pp. 213–18. New York: ACM.

Cogdill, K. (1999) *MEDLINEplus Interface Evaluation: Final Report*. Unpublished Report, Human-Computer Interaction Lab (HCIL), University of Maryland, URL (consulted January 2006): http://www.wam.umd.edu/~kcogdill/medlineplus.htm

Cuomo, D.L. and Bowen, C.D. (1994) 'Understanding Usability Issues Addressed by Three User-system Interface Evaluation Techniques', *Interacting with Computers* 6 (1): 86–108.

Desurvire, H., Kondziela, J. and Atwood, M. (1992) 'What is Gained and Lost when Using Evaluation Methods other than Empirical Testing', in A. Monk, D. Diaper and M.D. Harrison (eds) *Proceeding of the HCI '92: People and Computers VII*, 15–18 September, York, UK, pp. 89–102. Cambridge: Cambridge University Press.

Doubleday, A., Ryan, M., Springett, M. and Sutcliffe, A. (1997) 'A Comparison of Usability Techniques for Evaluating Design', in *DIS '97: Proceedings of the Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, 18–20 August, Amsterdam, The Netherlands, pp. 101–10. New York: ACM.

Fenichel, C.H. (1981) 'Online Searching Measures that Discriminate among Users with Different Types of Experience', *Journal of the American Society for Information Science* 32 (1): 23–32.

Frants, V.I., Shapiro, J., Taksa, K. and Voiskunskii, V.G. (1999) 'Boolean Search: Current State and Perspectives', *Journal of the American Society for Information Science* 50 (1): 86–95.

Hsieh-Yee, I. (1993) 'Effects on Search Experience and Subject Knowledge on the Search Tactics of Novice and Experienced Searchers', *Journal of the American Society for Information Science* 44 (3): 161–74.

Jeffries, R., Miller, J.R., Wharton, C. and Uyeda, K.M. (1991) 'User Interface Evaluation in the Real World: A Comparison of Four Techniques', in *CHI '91: Proceedings of the Conference on Human Factors in Computing Systems – Reaching through Technology*, 27 April–2 May, New Orleans, LA, pp. 119–24. New York: ACM.

Marchionini, G. and Komlodi, A. (1998) 'Design of Interfaces for Information Seeking', *Annual Review of Information Science and Technology* 38, pp. 89–130, Medford, NJ: Information Today.

Meyyappan, N., Chowdhury, G.G. and Foo, S. (2001) 'Use of a Digital Work Environment Prototype to Create a User-centred University Digital Library', *Journal of Information Science* 27 (4): 249–64.

Meyyappan, N., Foo, S. and Chowdhury, G.G. (2004) 'Design and Evaluation of a Task-based Digital Library for the Academic Community', *Journal of Documentation* 60 (4): 449–75.

Morgan, E.L. (1999) 'MyLibrary@NCState: The Implementation of a User-centered, Customizable Interface to a Library's Collection of Information Resources', in *SIGIR'99: 22nd International Conference on Research and Development in Information Retrieval*, 15–19 August, University of California, Berkeley, URL (consulted January 2006): http://dewey.library.nd.edu/mylibrary/sigir-99/

Mulhem, P. and Nigay, L. (1996) 'Interactive Information Retrieval Systems: From User Centered Interface Design to Software Design', in *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 18–22 August, Zurich, Switzerland, pp. 326–34. New York: ACM.

Nielsen, J. (1990) 'Paper Versus Computer Implementations as Mockup Scenarios for Heuristic Evaluation', in *IFIP INTERACT '90: Proceedings of the 3rd International Conference on Human-Computer Interaction*, 27–31 August, Cambridge, UK, pp. 315–20. Amsterdam: North Holland.

Nielsen, J. (1992) 'Finding Usability Problems through Heuristic Evaluation', in *CHI '92: Proceedings of the Conference on Human Factors in Computing Systems*, 3–7 May, Monterey, CA, pp. 373–80. New York: ACM.

Nielsen, J. (1993) 'Iterative User Interface Design', *IEEE Computer* 26 (11): 32–41.

Nielsen, J. (1994) 'Heuristic Evaluation', in J. Nielsen and R.L. Mack (eds) *Usability Inspection Methods*, pp. 25–62. New York: John Wiley.

Peng, L.K., Ramaiah, C.K. and Foo, S. (2004) 'Heuristic-based User Interface Evaluation at Nanyang Technological University in Singapore', *Program: Electronic Library and Information Systems* 38 (1): 42–59.

Penhale, S.J. and Taylor, N. (1986) 'Integrating End-user

Searching into a Bibliographic Instruction Program', *RQ* 27 (2): 212–20.

Petrelli, D., Hansen, P., Beaulieu, M., Sanderson, M., Demetriou, G. and Herring, P. (2004) 'Observing Users – Designing Clarity: A Case Study on the User-centred Design of a Cross-language Retrieval System', *Journal of the American Society for Information Science and Technology* 55 (10): 923–34.

Rudd, J., Stern, K. and Isensee, S. (1996) 'Low vs. High-fidelity Prototyping Debate', *Interactions* 3 (6): 76–85.

Shneiderman, B. (1998) *Designing User Interface: Strategies for Effective Human-Computer Interaction*. Reading, MA: Addison-Wesley.

Spyridakis, J.H. (1992) 'Conducting Research in Technical Communication: The Application of True Experimental Design', *Technical Communication* 39 (4): 607–24.

Sullivan, M.V., Borgman, C.L. and Wippern, D. (1990) 'End-user, Mediated Searches, and Front-end Assistance Programs on Dialog: A Comparison of Learning, Performance and Satisfaction', *Journal of the American Society for Information Science* 41 (1): 27–42.

Sutcliffe, A., Ennis, M. and Watkinson, S.J. (2000) 'Empirical Studies of End-user Information Searching', *Journal of the American Society for Information Science* 51 (13): 1211–31.

Van House, N.A., Butler, M.H., Ogle, V. and Schiff, L. (1996) 'A User-centered Iterative Design for Digital Libraries: The Cypress Experience', *D-Lib Magazine*, URL (consulted January 2006): http://www.dlib.org/dlib/february96/02vanhouse.html

Virzi, R.A. (1992) 'Refining the Test Phase of Usability Evaluation: How many Subjects is Enough?', *Human Factors* 34 (4): 457–68.

Virzi, R.A., Sokolov, J.L. and Karis, D. (1996) 'Usability Problems Identification Using both Low- and High-fidelity Prototypes', in *CHI '96: Proceedings of the conference on Human Factors in Computing Systems*, 13–18 April, Vancouver, Canada, pp. 236–243. New York: ACM.

Zhang, X. and Chignell, M. (2001) 'Assessment of the Effects of User Characteristics on Mental Models of Information Retrieval Systems', *Journal of the American Society for Information Science* 52 (5): 445–59.